Problem statement: List the component of Hadoop 2.x

Hadoop 2.x has following three major components.

HDFS

Yarn

MapReduce

HDFS 2.x Daemons: Name Node, Secondary Name Node (not required in HA) and Data Nodes

**Name Node**: NameNode is the master node in the Apache Hadoop HDFS Architecture that maintains and manages the blocks present on the DataNodes (slave nodes). NameNode is a very highly available server that manages the File System Namespace and controls access to files by clients.

**Data** **Node**: DataNodes are the slave nodes in HDFS. Unlike NameNode, DataNode is commodity hardware, that is, a non-expensive system which is not of high quality or high-availability.

Functions of DataNode:

These are slave daemons or process which runs on each slave machine.

The actual data is stored on DataNodes.

The DataNodes perform the low-level read and write requests from the file system’s clients.

They send heartbeats to the NameNode periodically to report the overall health of HDFS, by default, this frequency is set to 3 seconds.

**Secondary Name Node**: Third daemon or a process called Secondary NameNode

The Secondary NameNode is one which constantly reads all the file systems and metadata from the RAM of the NameNode and writes it into the hard disk or the file system.

It is responsible for combining the EditLogs with FsImage from the NameNode.

It downloads the EditLogs from the NameNode at regular intervals and applies to FsImage. The new FsImage is

copied back to the NameNode, which is used whenever the NameNode is started the next time.

Hence, Secondary NameNode performs regular checkpoints in HDFS. Therefore, it is also called CheckpointNode.

**Yarn : Yet another resource negotiator .**

In YARN there are three processes.

the Job Submitter (the client)

the Resource Manager (the master)

the Node Manager (the slave)

The application startup process is the following:

1 a client submits an application to the Resource Manager

2 the Resource Manager allocates a container

3 the Resource Manager contacts the related Node Manager

4 the Node Manager launches the container

5 the Container executes the Application Master

**Mapreduce**:

MapReduce is a programming framework that allows us to perform distributed and parallel processing on large data sets in a distributed environment.

MapReduce consists of two distinct tasks – Map and Reduce.

As the name MapReduce suggests, reducer phase takes place after mapper phase has been completed.

So, the first is the map job, where a block of data is read and processed to produce key-value pairs as intermediate outputs.

The output of a Mapper or map job (key-value pairs) is input to the Reducer.

The reducer receives the key-value pair from multiple map jobs.

Then, the reducer aggregates those intermediate data tuples (intermediate key-value pair) into a smaller set of tuples or key-value pairs which is the final output.

